CLAIMS

We claim as follows:

1. A computer implemented method for building a financial transfer model, comprising:

(a)   presenting to a user a screen interface having a tool pallet with one or more icon tools and a workspace for creating a graphical representation of the financial transfer model, the one or more icon tools including at least one tool for generating financial transfer activity icon instances upon the workspace with each instance having at least one attribute for defining an associated transaction between a payer and a payee entity;

(b)   receiving commands from the user for building upon the workspace the graphical representation of the financial transfer model including receiving commands for (i) generating one or more financial transfer activity icon instances upon the workspace (ii) associating at least one financial transfer transaction between a payer and a payee entity for each activity icon instance, and (iii) inter-connecting the one or more icon instances to generate the graphical financial transfer model; and

(c)   conveying instructions to an application component responsive to receiving said user commands for defining a financial transfer model data structure that corresponds to the graphical financial transfer model.

2. The method of claim 1, wherein the act of providing a tool pallet with one or more financial transfer activity icon tools includes providing a recurring and a non recurring financial transfer activity icon tool.

3. The method of claim 1, wherein the act of providing a tool pallet includes providing a pallet with one or more tools for creating relational connections between financial transfer activity

icon instances, said tools including information and timer flow tools for creating financial

transfer and timing relationship connections.

4. The method of claim 3, wherein the act of providing a tool pallet includes providing a timer icon tool for creating a timer icon instance that can be configured between a first and second activity for creating a time delay relationship between said first and second activities.

5. The method of claim 1, wherein the financial transfer model data structure is created with object class instances corresponding to financial transfer activity icon instances defined by the user.

6. The method of claim 5, wherein the object instances are defined in a directed graph data structure.

7. The method of claim 6, wherein the act of providing a tool pallet includes providing a pallet with a start icon tool for generating a start icon instance for the beginning of the graphical financial transfer model, wherein the start icon instance corresponds to a root object instance in the directed graph data structure.

8. The method of claim 1, wherein the act of conveying instructions to an application component responsive to receiving said user commands for defining a financial transfer model data structure that corresponds to the graphical financial transfer model includes

conveying said instructions to modify the data structure model if the received user commands comply with predefined syntax rules.

9. The method of claim 8, further comprising notifying the user that a received command for building the graphical financial transfer model is not allowed and not processing the command if the received command is not in compliance with the predefined syntax rules.

10. The method of claim 1, further comprising invoking a completeness checking routine to determine if the financial transfer data structure model is complete based on predefined criterion in response to a request from the user.

11. A memory storage media having machine-readable instructions for implementing a financial transfer editor program for building a financial transfer model, the instructions when executed causing a computer to perform a method comprising:

(a)   presenting to a user a screen interface having a tool pallet with one or more icon tools and a workspace for creating a graphical representation of the financial transfer model, the one or more icon tools including at least one tool for generating financial transfer activity icon instances upon the workspace with each instance having at least one attribute for defining an associated transaction between a payer and a payee entity;

(b)   receiving commands from the user for building upon the workspace the graphical representation of the financial transfer model including receiving commands for (i) generating one or more financial transfer activity icon instances upon the workspace (ii) associating at least one financial transfer transaction between a payer and a payee entity for each activity icon

instance, and (iii) inter-connecting the one or more icon instances to generate the graphical

financial transfer model ; and

(c)   responsive to receiving said user commands, defining a financial transfer model

data structure that corresponds to the graphical financial transfer model.

5

12. The memory storage media of claim 11, wherein the act of providing a tool pallet with one or

more financial transfer activity icon tools includes providing a recurring and a non recurring

financial transfer activity icon tool.

13. The memory storage media of claim 11, wherein the act of providing a tool pallet includes

providing a pallet with one or more tools for creating relational connections between

financial transfer activity icon instances, said tools including information and timer flow

tools for creating financial transfer and timing relationship connections.

14. The method of claim 11, wherein the financial transfer model data structure is created with

object class instances corresponding to financial transfer activity icon instances defined by

the user.

15. The memory storage media of claim 14, wherein the object instances are defined in a directed

20   graph data structure.

16. The memory storage media of claim 15, wherein the act of providing a tool pallet includes

providing a pallet with a start icon tool for generating a start icon instance for the beginning

of the graphical financial transfer model, wherein the start icon instance corresponds to a root

object instance in the directed graph data structure.

17. A memory storage media having machine-readable instructions for implementing a financial

transfer editor program for building a financial transfer model, the instructions when

executed causing a computer to perform a method comprising:

(a)   presenting to a user one or more icon tools for creating a graphical representation

of a financial transfer model, the one or more icon tools including at least one tool for generating

a financial transfer activity icon instance with each instance having at least one attribute for

defining an associated transaction between a payer and a payee entity;

(b)   receiving commands from the user for building the graphical representation of the

financial transfer model including receiving commands for (i) generating one or more financial

transfer activity icon instances upon the workspace (ii) associating at least one financial transfer

transaction between a payer and a payee entity, and (iii) inter-connecting the one or more icon

instances to generate the graphical financial transfer model ; and

(c)   responsive to receiving said user commands, defining a financial transfer model

data structure that corresponds to the graphical financial transfer model.

18. The memory storage media of claim 17, wherein the machine readable instructions include a

plurality of instructions corresponding to a database management component that defines the

financial transfer model data structure in response to the received commands from the user.

19. The memory storage media of claim 18, wherein the instructions comprise a plurality of instructions corresponding to a financial transfer modeling editor component that invokes the database management component to define the financial transfer model data structure in response to receiving the user commands via a plurality of instructions that correspond to a

5   graphical user interface component.

20. The memory storage media of claim 18, wherein the database management component is an object oriented database application program.